

**Amendments To the Claims**

1. (Currently Amended) A method comprising:

identifying a plurality of states and associated state classes to a state machine;

identifying a plurality of events and associated state transitions to the state machine; and

the state machine creating state objects and a transition map according to the plurality of

states and events; and

identifying at least one plug-in class to the state machine, the plug-in class configured to

monitor predetermined events.

~~2. (Cancelled) The method of claim 1 further comprising:~~

~~identifying at least one plug-in class to the state machine, the plug-in class configured to monitor~~

~~predetermined events.~~

3. (Currently Amended ) The method of claim 2 1 further comprising:

the state machine creating a plug-in object according to the plug-in class, the plug-in

object interacting with the transition map to cause state transitions.

4. (Original) The method of claim 1 further comprising:

identifying at least one state factory to the state machine, the state machine invoking the

state factory to create the state objects.

1 5. (Original) The method of claim 1 further comprising:

2 identifying at least one plug-in factory to the state machine, the state machine invoking  
3 the plug-in factory to create the plug-in objects.  
4

1 6. (Currently Amended) A method comprising:

2 extending a base state class to create at least one extended state class;  
3 configuring a base state machine class to operate with the extended state class;  
4 configuring the base state machine class to cooperate with a state factory class to create  
5 state objects according to the extended state class; and  
6 associating the extended state class with a state; and  
7 extending a base plug-in class to create at least one extended plug-in class;  
8 configuring the base state machine class to operate with the extended plug-in class; and  
9 associating an event monitored by the extended plug-in class with a state transition.  
10

1 ~~7. (Cancelled) The method of claim 6 further comprising:~~

2 ~~extending a base plug-in class to create at least one extended plug-in class;~~  
3 ~~configuring the base state machine class to operate with the extended plug-in class; and~~  
4 ~~associating an event monitored by the extended plug-in class with a state transition.~~  
5

1 8. (Previously Cancelled).

1 9. (Currently Amended) The method of claim 76 in which the base state machine class is  
2 configured to cooperate with a plug-in factory class to create plug-in objects according to the  
3 extended plug-in class.

4  
1 10. (Currently Amended) An article comprising:

2 a machine-readable medium comprising instructions which, when executed by a  
3 processor, result in:

4 identifying a plurality of states and associated state classes to a state machine;  
5 identifying a plurality of events and associated state transitions to the state machine; and  
6 the state machine creating state objects and a transition map according to the plurality of  
7 states and events; and  
8 identifying at least one plug-in class to the state machine, the plug-in class configured to  
9 monitor predetermined events.  
10

1 11. (Cancelled) ~~The article of claim 10 further comprising instructions which, when executed by~~  
2 ~~the processor, result in:~~  
3 ~~identifying at least one plug-in class to the state machine, the plug-in class configured to monitor~~  
4 ~~predetermined events.~~  
5

1 12. (Currently Amended) The article of claim ~~11~~12 further comprising instructions which, when  
2 executed by the processor, result in:

3 the state machine creating a plug-in object according to the plug-in class, the plug-in  
4 object interacting with the transition map to cause state transitions.

1 13. (Original) The article of claim 10 further comprising instructions which, when executed by  
2 the processor, result in:

3 identifying at least one state factory to the state machine, the state machine invoking the  
4 state factory to create the state objects.

1 14. (Original) The article of claim 10 further comprising instructions which, when executed by  
2 the processor, result in:

3 \_\_\_\_\_ identifying at least one plug-in factory to the state machine, the state machine invoking  
4 the plug-in factory to create the plug-in objects.

1 15. (Currently Amended) A system comprising:

2 a processor; and

3 a machine-readable medium comprising instructions which, when executed by the  
4 processor, result in;

5 identifying a plurality of states and associated state classes to a state machine;

6 identifying a plurality of events and associated state transitions to the state machine; and

7 the state machine creating state objects and a transition map according to the plurality of  
8 states and events; and  
9 identifying at least one plug-in class to the state machine, the plug-in class configured to  
10 monitor predetermined events.

11  
1 16. (Cancelled) ~~The system of claim 15 further comprising instructions which, when executed by~~  
2 ~~the processor, result in:~~  
3 ~~identifying at least one plug-in class to the state machine, the plug-in class configured to monitor~~  
4 ~~predetermined events.~~

5  
1 17. (Currently Amended) The system of claim 15<sup>6</sup> further comprising instructions which, when  
2 executed by the processor, result in:  
3 the state machine creating a plug-in object according to the plug-in class, the plug-in  
4 object interacting with the transition map to cause state transitions.

5  
1 18. (Original) The system of claim 15 further comprising instructions which, when executed by  
2 the processor, result in:  
3 identifying at least one state factory to the state machine, the state machine invoking the  
4 state factory to create the state objects.

5  
1 19. (Original) The system of claim 15 further comprising instructions which, when executed by  
2 the processor, result in:

- 3 identifying at least one plug-in factory to the state machine, the state machine invoking
- 4 the plug-in factory to create the plug-in objects.